



# Application de la méthode de faisceaux à la gestion journalière de la production

Claude Lemaréchal, Claudia Sagastizábal

## ► To cite this version:

Claude Lemaréchal, Claudia Sagastizábal. Application de la méthode de faisceaux à la gestion journalière de la production. [Rapport de recherche] RT-0184, INRIA. 1995, pp.16. inria-00069987

**HAL Id: inria-00069987**

**<https://inria.hal.science/inria-00069987>**

Submitted on 19 May 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

***Application of bundle methods to the  
unit-commitment problem***

C. Lemaréchal, C. Sagastizábal

**N° 0184**

Novembre 1995

PROGRAMME 5

Traitement du signal,  
automatique  
et productique

 ***rapport  
technique***

**1994**





# Application of bundle methods to the unit-commitment problem

C. Lemaréchal, C. Sagastizábal

Programme 5 — Traitement du signal, automatique et productique  
Projet Promath

Rapport technique n° 0184 — Novembre 1995 — 16 pages

**Abstract:** Optimizing the daily production in an electrical network (the so-called unit-commitment problem) is an heterogeneous large-scale problem (some 200 power plants in the French network). It lends itself particularly well to Lagrangian decomposition, which results in a nonsmooth optimization problem.

The present report concerns the application of a bundle method to this problem, following various decomposition schemes. This work was the object of a contract between the French Electricity Board (EdF) and Inria.

**Key-words:** unit-commitment problem, optimization, Lagrangian relaxation, bundle method

*(Résumé : tsvp)*

Rapport Final - Contrat EdF n° R31/1J3669/ER238 - octobre 1995

# **Application de la méthode de faisceaux à la gestion journalière de la production**

**Résumé :** L'optimisation de la production journalière d'électricité est un problème hétérogène de grande taille (quelque 200 groupes de production dans le réseau français). Il se prête particulièrement bien à la décomposition lagrangienne, ce qui donne lieu à un problème d'optimisation non différentiable.

Le présent rapport rend compte de la résolution de ce problème par application de la méthode de faisceaux, suivant différents schémas de dualisation. Ce travail a constitué un contrat entre EdF et l'Inria.

**Mots-clé :** gestion de la production électrique, optimisation, relaxation lagrangienne, méthode de faisceaux

## 1 Synthèse

Ce travail a été effectué en étroite collaboration avec F. Pellegrino et A. Renaud, du Département Méthodes d'Optimisation et de Simulation, Service Etudes de Réseaux, Direction des Etudes et Recherches d'EdF.

Le contrat avait comme finalité d'analyser la mise en œuvre de la méthode des faisceaux sur les problèmes de grande taille posés par la gestion journalière de la production, suivant une approche par relaxation lagrangienne. Il s'agissait de greffer le code N1CV2A, développé à l'Inria, avec le logiciel APOGEE d'EdF, et de comparer avec son aide deux schémas de décomposition (respectivement spatiale et spatio-temporelle).

Ces objectifs ont été atteints, les interrogations posées au départ ont été levées d'une façon positive. Une raison importante du bon déroulement de notre travail a été l'étroite et constante collaboration entre les chercheurs impliqués, aussi bien d'EdF que de l'INRIA; en particulier, le nouveau logiciel APOGEE apparaît comme un excellent produit. Nos résultats ont fait l'objet d'une présentation à un congrès et d'une publication, [3].

Une première conclusion de notre étude (voir § 4) est que nous disposons d'un logiciel de faisceaux robuste et précis. Il donne régulièrement 3 chiffres exacts dans la fonction duale sans trop d'efforts. Ceci confirme les résultats obtenus lors du CERD R34L01/1D6690. Il se confirme également que la méthode de faisceaux est économique en temps de calcul: l'essentiel d'une itération se passe dans la résolution des problèmes locaux d'APOGEE.

Par ailleurs, la décomposition spatio-temporelle apparaît comme tout à fait viable. Les problèmes que nous avons testés lui sont particulièrement défavorables, avec leur quelque 10000 variables duales, alors que la décomposition spatiale n'en implique que 48. Malgré cela, les qualités du code sont préservées; le prix à payer est un facteur 5 en temps d'exécution (et un facteur 10 en nombre d'itérations).

L'introduction de contraintes de réseau peut ainsi être envisagée avec optimisme. En effet, elles ne changent pas la complexité du problème dual, qui continuera à comporter  $10^4$  variables. On peut prévoir que la méthode de faisceaux continuera à converger en quelques centaines d'itérations. Par contraste, la décomposition purement spatiale deviendrait prohibitive, avec ses  $10^5$  variables.

En fait, le pour et le contre des deux schémas se résument comme suit:

	Spatial	Spatio-temp
sans réseau	+	−
avec réseau	− −	+
réduction saut dual		+?

Il reste que le temps total d'exécution, qui peut atteindre l'heure de SPARC 20, est un peu la pierre d'achoppement de cette approche. Pour gagner un ordre de grandeur dans les performances de l'algorithme dual, une révision substantielle de la méthode des faisceaux s'impose. Cela est envisageable en exploitant la structure additive de la fonction duale, voir § 6.

Mentionnons également que, grâce à des discussions conjointes très profitables, une réflexion de nature théorique a été engagée sur l'introduction des coupes supplémentaires permettant la réduction du saut de dualité; voir à nouveau § 6, un article est en préparation.

Dans le corps du rapport ci-dessous, nous commençons par un rappel du problème et des notations; ensuite, nous décrivons l'algorithme implémenté et toutes les routines de N1CV2A. Nous terminons avec une description détaillée des résultats, des difficultés rencontrées, et nous évoquons les perspectives issues de ce travail.

## 2 Description générale du problème

Considérons un problème de gestion journalière d'électricité. Soit  $I$  l'ensemble des unités de production, et soit  $T$  le nombre de pas de temps considérés. Notons  $p_i^t$  la production de l'unité  $i \in I$  au temps  $t$ . Alors,  $p^t := (p_1^t, p_2^t, \dots, p_{|I|}^t)$  représente la production totale du système à l'instant  $t$ ; de même  $p_i := (p_i^1, p_i^2, \dots, p_i^T)$  est la production de l'unité  $i$  pendant toute la période de temps considérée. Avec ces notations, le problème d'optimisation est le suivant:

$$\begin{cases} \min \sum_{i \in I} c_i(p_i) \\ p_i \in \mathcal{D}_i, & \text{pour tout } i \in I \text{ (en raccourci } p \in \mathcal{D}), \\ p^t \in \mathcal{S}^t, & \text{pour } t = 1, 2, \dots, T \text{ (en raccourci } p \in \mathcal{S}). \end{cases} \quad (1)$$

Ici,  $\mathcal{D} := \prod \mathcal{D}_i$  représente les contraintes dynamiques de la  $i$ -ème unité de production, et  $\mathcal{S} := \prod \mathcal{S}^t$  les contraintes statiques de demande et de transport. En fait, ces dernières contraintes sont linéaires et s'expriment comme

$$\text{pour } t = 1, \dots, T, \quad p^t \in \mathcal{S}^t \Leftrightarrow A^t p^t = \sum_i A_i^t p_i^t = b^t. \quad (2)$$

Par ailleurs,  $c_i$  note le coût de production de l'unité  $i$  pendant toute la période de temps.

La décomposition par relaxation lagrangienne est un outil bien adapté, utilisé depuis longtemps. L'approche classique, par **décomposition spatiale**, consiste à dualiser les contraintes de demande et de réserve. Pour  $\mu = (\mu^1, \dots, \mu^T)$ , on définit le Lagrangien

$$L_\Sigma(p, \mu) := \sum_{i \in I} [c_i(p_i) - \mu^\top A_i p_i] + \mu^\top b.$$

La fonction duale associée est alors

$$\Sigma(\mu) := \min_{p \in \mathcal{D}} L_\Sigma(p, \mu) = \mu^\top b + \min_{p \in \mathcal{D}} \sum_{i \in I} [c_i(p_i) - \mu^\top A_i p_i] \quad (3)$$

et peut être calculée de façon décomposée: chaque usine  $i$  résout

$$\min \{c_i(p_i) - \mu^\top A_i p_i : p_i \in \mathcal{D}_i\}. \quad (4)$$

Notre objectif était d'opposer cette technique à la **décomposition spatio-temporelle**, qui est en fait la seule envisageable lorsque les contraintes de réseau sont prises en compte (de l'ordre de  $10^5$  lignes dans (2)). Formulons (1) en dédoublant artificiellement la variable  $p$ :

$$\min \sum_{i \in I} c_i(p_i) \quad \text{avec } p \in \mathcal{D}, \quad q \in \mathcal{S}, \quad p = q. \quad (5)$$

Un multiplicateur  $\lambda_i^t$  affecté à chacune des contraintes couplantes  $p_i^t = q_i^t$  donne le Lagrangien

$$L_\Psi(p, q, \lambda) := \sum_{i \in I} c_i(p_i) - \lambda p + \lambda q.$$

La fonction duale associée est maintenant

$$\Psi(\lambda) := \min_{p \in \mathcal{D}, q \in \mathcal{S}} L_\Psi(p, q, \lambda) = \min_{p \in \mathcal{D}} \sum_i [c_i(p_i) - \lambda_i p_i] + \min_{q \in \mathcal{S}} \sum_t \lambda^t q^t, \quad (6)$$

clairement décomposable via

$$\min \{c_i(p_i) - \lambda_i p_i : p_i \in \mathcal{D}_i\} \quad \text{et} \quad \min \{\lambda^t q^t : q^t \in \mathcal{S}^t\}. \quad (7)$$



### 3 Méthode de faisceaux

Quelle que soit la décomposition choisie, le problème est maintenant de maximiser la fonction duale, qui est concave non différentiable. Cette maximisation est faite par une méthode de faisceaux: N1CV2A, décrite dans [4]. Nous décrivons son fonctionnement pour la décomposition spatio-temporelle, le cas de la décomposition spatiale s'en déduisant mutatis mutandis.

#### 3.1 Schéma algorithmique

Pour un prix  $\lambda$  donné, la résolution de (6) se fait dans APOGEE, qui calcule une paire optimale  $\bar{p}, \bar{q}$  (dépendant de  $\lambda$ ). On dispose ainsi de la valeur de la fonction duale et d'un sous-gradient:

$$\Psi(\lambda) = L_{\Psi}(\bar{p}, \bar{q}, \lambda) \quad \text{et} \quad \nabla_{\lambda} L_{\Psi}(\bar{p}, \bar{q}, \lambda) = \bar{q} - \bar{p} =: g(\lambda) \in \partial \Psi(\lambda).$$

Dans la description ci-dessous d'une itération courante, nous supposons que **nbun** telles valeurs sont disponibles, calculées aux points  $\lambda^1, \dots, \lambda^{\text{nbun}}$ . On dispose également d'un "centre du prox"  $\hat{\lambda}$  (initialisé sur le premier itéré), destiné à approcher le prix optimal.

Tout candidat  $\lambda^c$  à devenir centre du prox est solution unique du **maître-programme stabilisé**

$$\max_{\lambda} \left\{ \hat{\Psi}^{\text{nbun}}(\lambda) - \frac{1}{2t} |\lambda - \hat{\lambda}|^2 \right\}, \quad (8)$$

où  $\hat{\Psi}^{\text{nbun}}$  est l'approximation polyédrique habituelle de Dantzig-Wolfe:

$$\hat{\Psi}^{\text{nbun}}(\lambda) := \min_{k \leq \text{nbun}} \{ \Psi(\lambda^k) + \langle g(\lambda^k), \lambda - \lambda^k \rangle \},$$

et  $t > 0$  est un "coefficient de stabilisation". Le problème linéaire-quadratique (8) est résolu par le code QPDF2, écrit par K.C. Kiwiel, [2], interfacé par FQPROX.

La valeur optimale de (8) sert à quantifier le progrès espéré sur  $\Psi$ . Plus précisément, le nombre

$$\delta := \hat{\Psi}^{\text{nbun}}(\lambda^c) - \frac{1}{2t} |\lambda^c - \hat{\lambda}|^2 - \Psi(\hat{\lambda})$$

apparaît comme une “croissance nominale”. Pour promouvoir le candidat on exige une croissance suffisante sur la fonction duale:

$$\Psi(\lambda^c) \geq \Psi(\hat{\lambda}) + m_1 \delta. \quad (9)$$

La résolution de (8) fournit également un “gradient régularisé”  $G \in \partial \hat{\Psi}_n(\lambda^c)$  et une “ordonnée à l’origine”  $\varepsilon$ , tels que  $G \in \partial_\varepsilon \Psi(\hat{\lambda})$ .

Les blocs principaux d’une itération, coordonnés depuis N1CV2A, sont alors les suivants:

- 1) **Recherche curviligne** (NZSV1). Ajustement de  $t$  par essais et erreurs. À chaque cycle de cette routine, un choix est fait entre les quatre actions suivantes:

- $t$  est trop petit: (9) a lieu, mais

$$t \langle g(\lambda^c), G \rangle \geq m_E \delta.$$

Alors on résout (8) avec un coefficient  $t$  plus grand.

- Pas sérieux: (9) a lieu avec  $t$  suffisamment grand, on finit la recherche curviligne, le candidat est bon.
- $t$  est trop grand: (9) n’a pas lieu, et

$$\Psi(\lambda^c) + \langle g(\lambda^c), \hat{\lambda} - \lambda^c \rangle - \Psi(\hat{\lambda}) > m_I \delta.$$

On résout (8) avec un coefficient  $t$  plus petit. Note:  $m_E$  et  $m_I$  sont des paramètres fixés, tout comme  $m_1$ .

- Pas nul: (9) n’a pas lieu et  $t$  est suffisamment petit. On finit la recherche curviligne, le modèle n’est pas suffisamment bon.

Naturellement, chaque cycle implique la résolution de (8) dans FQPROX, suivie d’une résolution des problèmes locaux donnant  $\Psi(\lambda^c)$  et  $g(\lambda^c)$ .

- 2) **Gestion du faisceau** (FLIMIN, FREDF3). Prise en compte des limitations de mémoire. Le nombre **nbun** d’éléments de faisceau est limité à **memax**. Lorsque **nbun=memax**, au moins un élément est effacé; il est éventuellement remplacé par  $G$ .
- 3) **Actualisation du modèle**. Le nouvel élément  $g(\lambda^c)$  est ajouté au faisceau. En cas de pas sérieux, le centre du prox est transféré en  $\lambda^c$  et  $t$  est recalculé dans FPOVQN, suivant une formule de “quasi-Newton du pauvre”, [5].

- 4) **Préparation de l'itération suivante** (FREM2, FQPROX). Construction de la matrice du problème quadratique (8) et génération d'un nouveau candidat  $\lambda^c$ , préparant l'appel à NZSV1.
- 5) **Test d'arrêt** (FSTOP). Une fois (8) résolu, donnant  $G \in \partial_\varepsilon \Psi(\hat{\lambda})$ , on écrit:

$$\Psi(\lambda) \leq \Psi(\hat{\lambda}) + \langle G, \lambda - \hat{\lambda} \rangle + \varepsilon \leq \Psi(\hat{\lambda}) + |G| |\lambda - \hat{\lambda}| + \varepsilon. \quad (10)$$

Connaissant deux tolérances d'arrêt  $\varepsilon_0$  et  $\eta$ , l'algorithme se termine lorsque  $\varepsilon \leq \varepsilon_0$  et  $|G| \leq \eta$ .

### 3.2 Schéma informatique

L'interface entre APOGEE et la méthode des faisceaux est assurée par PROX-MYQN, dont la liste d'appel est donnée ci-dessous. Pour les arguments `nl`, `liste`, voir § 5.

```
subroutine proxmyqn(n,point,g,fp,dxmin,df1,epsrel,eta,zero,
  imp,io,mode,nbun,iter,nsim,memax,iz,niz,rz,nrz,dz,ndz,
  ite,criter,indic,nl,liste,xn,fn,ggn,alam,bundle)
```

```
n: nombre de variables (dimension de l'espace)
point: tableau de dimension n contenant la valeur courante des
       variables duales.
g: valeur d'un sous-gradient (entree) (tableau de dimension n)
fp: valeur de la fonction duale
dxmin: pouvoir de resolution (deux points distants de moins de
      dxmin sont "confondus" dans la recherche curviligne)
valeur typique: 10**-10
      (mais peut croître si l'on a un doute sur le sous-gradient)
df1: (0.1*fp marche en general) estimation de la marge a gagner
      sur la fonction duale (ne sert qu'a la premiere iteration)
epsrel: precision relative sur la valeur de la fonction
eta: precision absolue sur la violation des contraintes.
zero: precision de la machine (10**-12)
```

imp: parametre de commande des impressions de sortie. Permet  
par exemple de tabuler la recherche curviligne  
==0 rien n'est imprime.  
>=1 debut et fin.  
>=2 information a chaque reduction du faisceau.  
>=3 information a chaque iteration.  
>=4 detail sur la recherche curviligne: valeur du pas, de  
la fonction test, de sa derivee.  
>=5 retour a l'usine, utile pour les cas desesperes.  
io: fichier d'impression (numero logique)  
mode:=0 stop demande par l'utilisateur  
(indic=0 voir description d'indic)  
=1 sortie normale  
=2 erreur donnees initiales  
=3 non utilise  
=4 nombre d'iterations maximum atteint  
=5 nombre maximum d'appels a APOGEE atteint  
=6 echec: dxmin atteint (gradient peut-etre faux)  
=7,8 non utilises  
=9 memax trop petit, steepest descent arrive.  
<0 indic<0 rend la recherche curviligne malade.  
nbun: nombre d'elements total dans le faisceau quand CRITER=1  
iter: entree: nombre d'iterations maximal (grandes iterations)  
sortie: nombre d'iterations reel  
nsim: entree: nombre maximal d'appels a APOGEE  
sortie: nombre reel de minimisations sur le primal  
memax: nombre maximal d'elements du faisceau  
iz: tableau de travail entier  
niz: dimension du tableau (au moins memax)  
rz: premier tableau de travail double precision  
nrz: dimension du tableau rz (au moins 3\*n)  
dz: deuxieme tableau de travail double precision  
ndz: dimension du tableau dz  
(au moins 10+memax\*(3\*memax+17)/2)  
ite: numero de l'appel courant (nombre d'appels d'APOGEE)

criter: =0 continuer;  
          =1 c'est fini;  
          =2 nous vous autorisons a changer la liste;  
              dans ce dernier cas, faites criter=3 si vous  
              voulez le gradient regularise complet;  
          =3 nous avons calcule le gradient complet et  
              vous pouvez choisir une nouvelle liste  
              faites criter=4 apres.  
          =4 vous avez change la liste et nous continuons;

REMARQUE: possibilite non exploite par APOGEE:

indic: parametre de sortie pour permettre  
      des reponses anormales d'apogee  
      >0 retour normal, les calculs demandes ont ete  
          correctement effectues;  
      <0 les calculs n'ont pu etre effectues, soit parceque  
          dans la recherche curviligne le pas est trop grand,  
          soit parceque le gradient n'est pas borne, ou encore  
          parcequ'il y a des contraintes impossibles a satisfaire  
          dans le calcul de f;  
      =0 l'utilisateur desire arreter l'optimisation, soit  
          parceque le probleme est non borne ou parceque l'itere  
          courant lui semble optimal.

nl: dimension de la liste (<=n nombre de variables actives)

liste: tableau entier; liste des indices sur lesquels porte  
      l'optimisation duale

xn: tableau double precision de dimension n  
      le point de sortie (a peu pres le meilleur point trouve)

fn: la valeur en xn (dito: a peu pres la meilleure valeur)

gradreg: tableau double precision de dimension n  
          le gradient regularise en xn

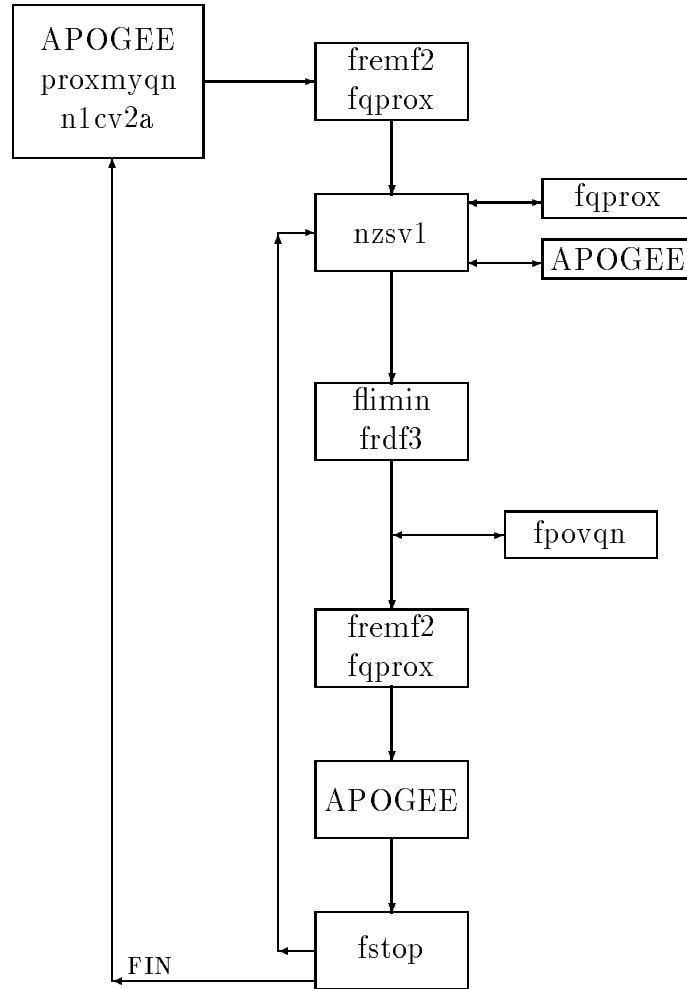
alam: tableau double precision de dimension memax

      multiplicateurs convexes qui forment le gradient regularise

bundle: tableau double precision de dimension memax\*n

le faisceau, avec les gradients stockés par colonnes.

De tous ces paramètres, seul **eta** est un peu délicat à régler; voir commentaires §5. Le déroulement d'une exécution se fait selon l'organigramme suivant:



## 4 Résultats numériques

Nous donnons les résultats sur 6 jeux de données, correspondant à six journées différentes. L’horizon  $T$  est de 48 demi-heures. Le réseau n’est pas pris en compte (la réserve non plus; il y a 48 lignes dans (2)). Les exemples 1-4 ont 207 groupes en production, tandis que les exemples 5-6 en ont 226. Cela donne donc respectivement  $48 \times 207 = 9936$  et  $48 \times 226 = 10848$  variables duales pour la décomposition spatio-temporelle  $\Psi$  (alors que la décomposition spatiale  $\Sigma$  a dans tous les cas 48 variables duales). Le tableau de résultats compare le nombre d’itérations requis par chaque technique de décomposition. La première colonne donne le nombre de résolutions des problèmes locaux, la deuxième colonne la valeur duale obtenue, et la troisième, la norme du gradient régularisé final. Pour la colonne “Partiel”, voir § 5. Les deux dernières colonnes sont consacrées aux temps de calcul: le temps total en secondes sur un SPARC 20, et la proportion passée dans la méthode des faisceaux elle-même (le complément étant donc le temps utilisé par les problèmes locaux).

Test	# iter	Valeur duale	$\ G\ $	Partiel	N1CV2A	CPU(s)
1 $\Sigma$	33	5195575	150		0.02%	351
1 $\Psi$	250	5194556	100	85	22%	1490
2 $\Sigma$	19	7842647	700		0.01%	327
2 $\Psi$	98	7841180	150	41	11%	718
3 $\Sigma$	25	6499580	160		0.02%	237
3 $\Psi$	280	6499297	130	60	27%	1480
4 $\Sigma$	20	8310985	690		0.02%	238
4 $\Psi$	131	8310235	160	37	14%	590
5 $\Sigma$	58	4153291	110		0.04%	480
5 $\Psi$	561	4152531	70	209	31%	3880
6 $\Sigma$	34	5679565	160		0.03%	326
6 $\Psi$	216	5678200	110	83	22%	1480

## 5 Les difficultés rencontrées

Un phénomène courant en optimisation non différentiable (et en particulier avec la méthode de faisceaux) est que la valeur optimale du critère est assez rapidement obtenue, la difficulté étant ensuite de s'assurer que cette valeur est bien optimale. Autrement dit, activer le test d'arrêt (10) avec  $|G|$  et  $\varepsilon$  tout les deux petits exige beaucoup d'itérations. Cette difficulté est illustrée par la colonne "Partiel" du tableau des résultats: elle donne le nombre d'itérations nécessaire pour obtenir 3 chiffres exacts dans la valeur duale.

À cette difficulté vient s'ajouter le choix de  $\eta$  pour le test d'arrêt, qui s'avère d'autant plus délicat que la taille du problème est grande. En effet, écrivons (10) pour un  $\lambda = \bar{\lambda}$  optimal. On voit que, pour avoir un bon équilibre à l'arrêt, les termes  $|G| |\bar{\lambda} - \hat{\lambda}|$  et  $\varepsilon$  doivent avoir des magnitudes comparables. Sachant qu'on souhaite obtenir  $\varepsilon \leq \varepsilon_0$ , où  $\varepsilon_0$  est fixé à une valeur connue (disons  $10^{-3}$  en valeur relative), il s'agit de trouver une valeur adéquate pour  $|G|$ , c'est-à-dire pour  $\eta$ . Soit  $d$  une estimation du maximum des coordonnées de  $\hat{\lambda} - \bar{\lambda}$ ; autrement dit,  $d$  représente "les limites de l'épure" autour de  $\hat{\lambda}$ , à l'intérieur desquelles au moins un jeu de prix raisonnables devrait exister. Alors  $|\bar{\lambda} - \hat{\lambda}|^2$  peut être majoré (grossièrement) par  $nd^2$ , où  $n$  est le nombre de variables duales. On en déduit la valeur correspondante  $\eta = \varepsilon_0/d\sqrt{n}$ . Le test d'arrêt est donc plus strict pour les problèmes de grande taille (décomposition  $\Psi$ ) que pour ceux de taille moyenne (décomposition  $\Sigma$ ).

Pour nos tests, nous avons choisi  $d = 0.1\text{FF/kW}$  et  $\varepsilon_0 = 0.1\%$ .

Pour lutter contre la dimensionalité, nous avons optimisé le calcul de la matrice du problème quadratique (8), formée des produits scalaires entre les sous-gradients du faisceau. Ce calcul et la résolution de (8) sont exécutés maintenant de façon très performante.

Nous avons aussi implémenté une variante visant à accélérer la réduction de  $|G|$  selon l'idée suivante. Certains groupes de production ont une influence très importante dans le déroulement des itérations, alors que d'autres participent peu au comportement de la fonction duale. On sélectionne donc les groupes les plus importants; ils sont transmis par l'argument `liste`, de `nl` éléments, dans la liste d'appel à `PROXMYQN`. La maximisation duale est alors restreinte à ces `nl` groupes, la liste étant révisée périodiquement. Cette technique a deux avantages: pour la décomposition spatio-temporelle, la maximisation duale se



fait dans un espace de dimension réduite; d'autre part, le nombre `n1` de groupes à optimiser par APOGEE est réduit.

Cette variante ne nous a malheureusement pas permis un gain de temps significatif: les économies ainsi réalisées n'étaient pas suffisantes pour compenser les calculs supplémentaires exigés par une gestion efficace des groupes importants.

## 6 Perspectives

**Implémentation.** Il peut être souhaitable de chercher à améliorer les performances du code de résolution duale. Pour cela, une piste très prometteuse existe, qui constitue un raffinement de la méthode de faisceaux (ou même de Dantzig-Wolfe pure), tenant compte de la structure additive du critère dual.

Très schématiquement, l'idée est la suivante. Pour un  $\lambda = \lambda^c$  donné, on utilise la valeur  $\Psi(\lambda^c)$  et un sous-gradient  $g(\lambda^c)$ :

$$\Psi(\lambda) \leq \Psi(\lambda^c) + \langle g(\lambda^c), \lambda - \lambda^c \rangle.$$

Or on a  $\Psi(\lambda) = \sum \Psi_i(\lambda)$  et  $g(\lambda) = \sum g_i(\lambda)$ ; APOGEE optimise chaque groupe séparément, et est donc capable de fournir des informations **désagrégées**  $\Psi_i(\lambda^c)$ ,  $g_i(\lambda^c)$ :

$$\Psi_i(\lambda) \leq \Psi_i(\lambda^c) + \langle g_i(\lambda^c), \lambda - \lambda^c \rangle,$$

ce qui donne l'approximation bien plus précise

$$\Psi(\lambda) \leq \sum_i [\Psi_i(\lambda^c) + \langle g_i(\lambda^c), \lambda - \lambda^c \rangle]$$

qui peut être insérée dans (8).

On sait que cette technique est extrêmement efficace; par exemple, elle améliore Dantzig-Wolfe de façon considérable.

**Analyse théorique.** L'optimisation de la production est un problème non convexe; en tant que tel, il est affecté par un *saut dual* entre les valeurs primale et duale. Ce saut de dualité peut être plus ou moins important suivant le schéma de dualisation adopté; on peut donc se poser des questions du type:  
– comment se comparent les sauts duaux, c'est-à-dire les valeurs duales optimales  $\Sigma(\bar{\mu})$  et  $\Psi(\bar{\lambda})$ ?

- peut-on améliorer le saut dual en jouant sur la définition des ensembles  $\mathcal{S}$  et  $\mathcal{D}$ ? Ceci ne vaudrait que pour la décomposition spatio-temporelle puisque, dans la décomposition spatiale, la structure des contraintes (2) est figée.
- comment interpréter les solutions duales optimales  $\bar{\lambda}$  comme prix marginaux associés à des perturbations du problème initial (1)?

Une réflexion dans ce sens a déjà été engagée, utilisant la théorie développée dans le Chapitre XII de [1].

## References

- [1] J.-B. HIRIART-URRUTY ET C. LEMARÉCHAL, *Convex Analysis and Minimization Algorithms*, Springer-Verlag, 1993.
- [2] K.C. KIWIEL, A method for solving certain quadratic programming problems arising in nonsmooth optimization. *IMA Journal of Numerical Analysis*, 6, pp. 137-152, 1986.
- [3] C. LEMARÉCHAL, F. PELLEGRINO, A. RENAUD ET C. SAGASTIZÁBAL, Bundle methods applied to the unit-commitment problem. Communication présentée à la 17ème Conférence IFIP TC7, Prague, juillet 1995. Acceptée pour publication dans les Actes de cette conférence.
- [4] C. LEMARÉCHAL ET C. SAGASTIZÁBAL, An approach to variable metric bundle methods. *Lecture Notes in Control and Information Sciences # 197, Systems Modelling and Optimization*, (éds. J. Henry et J-P. Yvon), Springer-Verlag, pp. 144-162, 1994.
- [5] C. LEMARÉCHAL, ET C. SAGASTIZÁBAL, Practical aspects of the Moreau-Yosida regularization II: Computational consequences, en préparation.

## Contents

<b>1</b>	<b>Synthèse</b>	<b>3</b>
<b>2</b>	<b>Description générale du problème</b>	<b>4</b>
<b>3</b>	<b>Méthode de faisceaux</b>	<b>6</b>
3.1	Schéma algorithmique . . . . .	6
3.2	Schéma informatique . . . . .	8
<b>4</b>	<b>Résultats numériques</b>	<b>12</b>
<b>5</b>	<b>Les difficultés rencontrées</b>	<b>13</b>
<b>6</b>	<b>Perspectives</b>	<b>14</b>



---

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,  
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY  
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex  
Unité de recherche INRIA Rhône-Alpes, 46 avenue Félix Viallet, 38031 GRENOBLE Cedex 1  
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex  
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

---

Éditeur

INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)

ISSN 0249-6399